

# Index



## Special Characters

- > redirection operator, 104
- < redirection operator, 104–105
- > redirection operator, 192
- 2> redirection operator, 105

## A

- @a scalar variable, 340–341
- Abs\_func.c file, 387–388
- Abs\_main.c file, 388
- Absolute.c file, 384
- abstract data type, 411
- algorithms, 256–262
- aliases, 210–211
- analyzing applications, 224–228
- A.out file, 293
- applets, 430
- applications
  - algorithms, 256–262
  - analyzing, 224–228
  - command execution in order of appearance, 193
  - conditional statement execution, 194–196
  - cutting and sorting fields, 165–167
  - decision logic, 194–196
  - design process, 179
  - designing, 158–160
  - designing records, 158–159

- different courses of action, 194–196
- extracting and displaying information, 164–165
- flowcharts, 224–227
- formatting output, 163–165
- linking files, 159
- logical structures, 158
- performing one of many actions, 201–202
- printing three fields, 163
- program development cycle, 179–182
- programmer file, 162
- programmer project assignments, 160–161
- project file, 162–163
- prototyping, 182
- pseudocode, 227–228
- record-searching capability, 245–246
- reusing code, 267–269
- running in background, 299–300
- sequential logic, 193
- shell programs, 167–170
- specifications, 179
- arguments
  - functions, 266
- arithmetic operators, 191–192, 372–374
  - precedence, 191
  - variables as operands, 191

- arrays, 329
  - of characters, 194
- ASCII text files, 102
- assemblers, 365
- assembly language, 364
- asterisk (\*) wildcard, 192
- automatic variables, 372
- automatically issuing commands, 178
- automatically loading functions, 268
- automatically removing temporary files, 212
- automatically starting X Window System, 422–423
- Awk, 131–133
- awk command, 131–133, 163–165
  - extracting and displaying information, 165
  - f option, 132
  - refining Vreport, 132–133
- Awk program
  - action statements, 132–133
  - BEGIN and END blocks, 332
  - comments (#), 332
  - f option, 332
  - FS variable, 132–133
  - implicit while loop, 331
  - versus* Perl, 330–333
  - running program prototype, 215–216
  - shortcut to file processing commands, 164
- Awkcom.a program, 330
- awsd.com Web site, 352

## B

- @b scalar variable, 340–341
- backquote (`) operator, 189
- Bash (Bourne Again Shell), 182–183
- Bash shell
  - environment and configuration variables, 185–187
- .bash\_profile file, 240
  - automatically loading functions, 268–269
- .bash\_profile login script
  - executing functions, 267
- .bashrc file, 211
- binary files, 102
- block special files, 103
- <BLOCKQUOTE> tag, 350
- </BLOCKQUOTE> tag, 350
- Boolean tests, 236–239
- /boot partition
  - verifying location, 285
- bootable floppy disk, 283–288
- branch instruction, 193
- Buildfile.c file, 386–387

## C

- C library, 367
- C programming language, 180, 364–365
  - accepting input, 393–396
  - arithmetic operators, 372–374
  - assemblers, 365
  - character constants, 370
  - comments (/ \* \*/), 368
  - compilers, 365

- constants, 369
- data types, 369–370
- debugging programs, 392–393
- declaring variables, 371–372
- decrement (--) operator, 373–374
- defining functions, 380–381
- do-while loop, 378–379
- encoding and decoding programs, 396–401
- executable files, 365
- files, 385–387
- functions, 365
- header files, 369
- identifiers, 371
- if statement, 377–378
- #include directive, 368–369
- increment (++) operator, 373–374
- key words, 365, 367
- linkers, 365
- loops, 378–379
- main function, 367
- make utility, 387–393
- object code, 365
- preprocessor, 365
- program creation, 365
- program format, 367
- source code, 365
- strings, 370–371
- variables, 369
- while loop, 378–379
- C++ programming language, 180, 405–406
  - CC compiler, 405
  - displaying message on-screen, 406
  - enhancing C functions, 409–410
  - function overloading, 405
  - functions, 405
  - methods, 406
  - objects, 406
  - reading text file, 407–409
  - setting up classes, 411–412
  - similarities to C programming language, 405
- Calendar application, 446
  - 12-hour time format, 449
  - Create New Appointment window, 450
  - Create To-Do Item window, 452
  - Day view, 447, 449
  - to do list, 447
  - entering appointments, 450–451
  - Month view, 448
  - Week view, 447
  - Year view, 449
- case logic, 179, 201–202
  - esac terminator, 202
- cat command, 167, 245
  - output redirection, 105
  - redirecting output, 113–114
- CC compiler, 405
- <CENTER> tag, 350
- </CENTER> tag, 350
- CGI (Common Gateway Interface), 351–357
  - INPUT TYPE=submit code, 351
- Chap1 file, 198
- char data type, 369
- character constants, 370
- character special files, 103
- [chars] wildcard, 192–193
- chmod command, 128, 181–182, 195
- class of characters, 192–193

- classes
  - setting up, 411–412
- clear command, 255–256
- clearing screen, 255–256
- clients, 419
- Clock applet, 431–432
- closing files, 385–386
- cmp command, 308
- COBOL, 180
- Colors script, 199
  - while loop test command usage, 242–243
- combining files, 112–115
  - delimiters, 115
- combining script files, 171–172
- comm command, 146–147, 166
- command line
  - executing control structures, 197
  - functions, 265, 267
  - test command, 229
- commands
  - aliases, 210–211
  - automatically issuing, 178
  - connecting several, 142
  - executing on signal, 212
  - execution in order of appearance, 193
  - exit status, 230–231
  - output as contents of variables, 189
  - redirecting input, 104
  - redirecting output to grep command, 144
  - redirecting output to input of another command, 142
- communication utilities, 280
- comparing files, 146–147, 308
- compilers, 181
  - C programming language, 365
- compiling and linking files, 387–388
- conditional statement execution, 194–196
- configuration variables, 183, 185–187
  - capitalization of, 184
- connecting several commands, 142
- constants, 326
  - beginning with 0, 192
  - C programming language, 369
- control string, 374
- control structures
  - executing at command line, 197
  - nested, 195–196
  - repeating, 196–201
- converting lowercase to uppercase characters, 151
- cookwood.com Web site, 352
- copying
  - files, 108–110
  - man page into /man directory, 313–314
- core files, 293
- Corp\_phone file
  - determining existence of, 239
  - previewing, 178
- Corp\_phones file
  - adding phone record, 217–218
  - creation of, 212–214
  - displaying contents, 261
  - repositioning cursor, 256
  - searching, 212
- cp command, 108–110
  - alias, 211
  - source files as arguments, 109
  - wildcards and, 109

- CPUload applet, 466–467
- Cube class, 411–412
- cube.C file, 411
- cursor
  - placing at data-entry point, 208–209
- cursor movement, 256
- Cursor script, 209
- customizing personal environment, 210–211
- cut command, 115–118, 164, 165–166
  - d option, 117
  - f option, 116
  - fixed-length records, 118
  - variable-length records, 118
- cutting and sorting fields, 165–167

## D

- data entry form
  - from while loop, 200–201
- data types
  - arrays, 329
  - C programming language, 369–370
  - constants, 326
  - hashes, 330
  - identifying, 325–330
  - numbers, 326
  - scalars, 326
  - strings, 326–327
  - variables, 326
- date command, 265
- datenow() function, 265
  - accepting argument, 266

- Datestuf.C file, 409–410
- dd command, 171, 282–283, 287–288
- debugging programs
  - C programming language, 392–393
- decision logic, 179, 194–196
  - if statement, 195–195
- Decode.c file, 398
- Decode\_file.c file, 398
- decrement (--) operator, 373–374
- decryption algorithm, 396
- defining and evaluating operators, 188–190
- deleting
  - comments from scripts, 171
  - files, 107–108
  - phone records, 246–250
- dependent files, 389
- designing applications, 158–160
- designing records, 158–159
- desktop
  - adjusting position of icons on panel, 466
  - configuring, 461–471
  - selecting and configuring screen-saver, 465–466
  - wallpaper, 461–465
- Desktop Area, 423
- devices
  - building file system on, 284
- df utility, 289–290
  - file system as argument, 289
  - h option, 290
- diamond operator (<>), 332, 336
- diff command, 147

- differences between files, 147
  - directories, 103
    - determining existence of, 239–241
    - locations of executable programs, 182
    - recursively removing, 110–111
    - removing, 108
  - Disk Usage applet, 466
  - documents
    - spell-checking, 306–308
  - DOG variable, 188
  - Dominion Consulting Web page
    - creation of, 358–361
  - double data type, 369
  - double quotes (“”) and spaces in variables, 188–189
  - do-while loop
    - C programming language, 378–379
  - du utility, 291–292
    - b option, 291
    - h option, 292
    - pipng output to more command, 291
  - duplicate lines
    - outputting, 145
    - removing from files, 144–145
  - duplicating files, 282–283
- ## E
- echo command, 188
    - n option, 194
  - empty files, 106
  - Encode.c file, 397, 459
  - Encode\_file.c file, 397
  - Encode\_make file, 399
  - encoding and decoding programs, 396–401
  - encrypted files, 396
  - environment variables, 183, 185–187
    - capitalization of, 184
    - creation of, 189–190
    - displaying contents of, 184
    - listing, 184
    - setting values, 184
  - equal sign (=) operator, 188
  - error messages
    - redirecting, 105
  - /etc/fstab file, 284
  - /etc/inittab file, 421–422
  - Example2.p program, 319
  - Example3.p program, 320
  - Example4.p program, 321
  - Example5.p program, 323
  - Example6.p program, 324–325
  - Example7.p program, 328
  - Example8.p program, 329
  - Example9.p program, 330
  - executable files, 181
    - C programming language, 365
  - executable programs
    - directory locations of, 182
  - executable script files, 128
  - executing commands
    - on signal, 212
  - exit status, 230–231
  - export command, 189–190
  - expressions
    - OR and AND relationships, 236–237
  - extracting and displaying information, 164–165

extracting information from two files  
  sharing field, 128–131  
extropia.com Web site, 352

## F

fail() function, 409  
fatal errors, 181  
\\fb command, 311  
fclose function, 386  
fdformat command, 284–285, 287  
feof function, 386  
fgetc function, 386  
\\fl command, 311  
fields  
  cutting, 165–167  
  printing, 163  
  sorting, 165–167  
File Manager  
  copying, renaming and deleting  
  files, 442–445  
  current working directory, 439  
  detailed listing, 439–440  
  directory creation, 445–446  
  directory tree, 437–438  
  home directory, 439  
  modifying file attributes, 440–442  
  running, 437–446  
file pointers, 385  
file processing, 102–106  
file processing utilities, 277–278,  
  282–294  
file structures, 103–104  
file system  
  building on device, 284  
  setting up, 285

filehandles, 333  
Fileread.C file, 407–409  
files  
  appending data, 112–114  
  appending text, 150–151  
  ASCII text, 102  
  back-up copies, 109  
  binary, 102  
  block special, 103  
  Boolean tests, 236–239  
  C programming language,  
  385–387  
  changing format of destination  
  copy, 282–283  
  character special, 103  
  closing, 385–386  
  combining, 112–115  
  comparing, 146–147, 308  
  compiling and linking, 387–388  
  copying, 108–110  
  counting number of lines, bytes, or  
  characters in, 148  
  creation of, 106–107  
  deleting, 107–108  
  determining existence of, 239–241  
  differences between, 147  
  directories, 103  
  displaying first few lines, 143  
  duplicating, 282–283  
  empty, 106  
  encrypted, 396  
  executable, 181, 235  
  finding, 112  
  flat ASCII, 102, 103  
  formatting output, 152–153  
  global changes to, 148–151

- input and output, 386
  - joining, 170
  - line-by-line combining, 114–115
  - linking, 159
  - manipulating, 106–123
  - modes, 181–182
  - moving, 111
  - opening, 385–386
  - ordinary, 102
  - Perl access of, 333–336
  - preventing redirection from overwriting, 192
  - readable, 236
  - regular, 102
  - removing columns or fields, 115–118
  - removing duplicate lines from, 144–145
  - removing erroneous character from, 152
  - renaming, 111
  - replacing characters in, 152
  - searching for pattern, 143–144
    - as series of characters, 102
  - sorting contents, 118–120
  - special, 103
  - testing, 234–236
  - testing existence, 234
  - testing for end of, 386
  - time and date stamp, 106–107
  - types, 102–103
  - validating existence, 241
  - viewing permissions, 234
  - warning before overwriting, 111
- find command, 112, 293
  - exec rm option, 293
  - o argument, 293
  - print option, 112
  - removing garbage files, 293–294
- finding files, 112
- fixed-length records, 104
  - cut command, 118
- flat ASCII files, 102, 103
- float data type, 369
- floppy disks
  - back-up copies, 287–288
  - formatting at low levels, 284–285
- flowcharts, 224–227
  - symbols, 226
- <FONT SIZE> tag, 350
- <FONT> tag, 350
- </FONT> tag, 350
- fopen function, 385–386
- for loops, 196–198
  - wildcards, 198
- FORM tag, 352
- format specifiers, 374–375
- formatted output, 374–377
- formatting
  - file output, 152–153
  - floppy disks at low levels, 284–285
- formatting output, 163–165
- formatting record output, 243–247
- formatting text, 309–314
- \\fP command, 311
- fputc function, 386
- \\fR command, 311
- free command, 298
  - forwarding display generated by, 298–299
- FS variable, 132–133
- function overloading, 405, 409–410
- function prototype, 381



- functions, 265–267, 267, 365, 367
  - accepting arguments, 384
  - arguments, 266, 381–382
  - automatically loading, 268
  - C++ programming language, 405
  - command line, 267
  - defining, 380–381
  - executing, 267
  - identifiers, 371
  - names, 265
  - return value, 384
  - return values, 383–384
  - shell scripts, 268
  - storing, 265

## G

- g++ compiler, 181, 406
- garbage files, 293–294
- gcc command, 376
- GET method, 352
- getline() function, 409
- glob characters, 192
- glob patterns, 192
- global changes to files, 148–151
- global variables, 372
- GNOME (GNU Network Object Model Environment), 420
- GNOME Pager, 430–432
  - Applications View, 431
  - Clock applet, 431–432
  - Desktop View, 430–431
- GNOME panel, 423, 428–432
  - adjusting position of icons, 466
  - CPUload applet, 466–467
  - GNOME Pager, 430–432
  - hiding, 432

- Main menu, 429
- Move applet, 466–467
- Phoneadd script, 468–469
- Terminal Emulator icon, 430
- Gnumeric Spreadsheet application, 453
  - cutting and pasting, 456–458
  - entering data, 454
  - numeric formatting, 454–455
- grep command, 143–144, 244
  - i option, 143
  - l option, 144
  - options, 143–144
  - redirecting command output to, 144
  - searching Corp\_phones file, 212
  - wildcards, 143–144
- groff command, 310
- GUI (graphical user interface), 418

## H

- <H1> tag, 350
- <H6> tag, 350
- hard disks
  - checking usage, 288–294
  - number of 1024-byte blocks allocated, 289–290
  - summarizing usage, 291–292
- hard drives
  - removing garbage files, 293–294
- hashes, 330
- head command, 143
- header files, 369
- help pop-up boxes, 427
- high-level languages, 180–181
  - storing statements, 181

\$HOME variable, 188–189  
HTML (Hyper Text Markup Language), 348  
  tags, 348  
HTML editors, 349  
<HTML> tag, 349  
</HTML> tag, 349  
hyperlinks, 348

## I

icons, 423  
identifiers, 371  
identifying data types, 325–330  
identifying records, 159  
if statement, 194–195  
  C programming language, 377–378  
  fi ending for, 195  
  test command usage by, 241  
if statements  
  nesting, 195–196  
if..else statement  
  fi ending for, 195  
inches.c file, 376–377  
#include directive, 368–369  
increment (++) operator, 373–374  
input  
  redirecting, 104  
input validation, 262–265  
int data type, 369  
integers  
  relational tests, 229–231  
interpreters, 181  
ispell utility, 306–308

## J

join command, 128–131, 166  
  -o option, 131  
  -a option, 131  
  -e option, 131  
  -j option, 130  
  -t option, 131  
joining files, 170

## K

Kernighan, Brian, 364  
key words  
  C programming language, 365, 367  
Keyboard.c file, 395–396  
key/value pairs, 330  
kill command, 296, 301–302

## L

less command, 214–215  
linkers, 365  
linking files, 159  
Linux  
  configuring to start X Window System, 421–422  
  g++ compiler, 181, 406  
  groff command, 310  
listing most CPU-intensive tasks, 295–297  
lname variable, 257  
localhost networking feature, 348  
logic errors, 181

- logic structures, 179, 193–203
  - case logic, 201–202
  - decision logic, 194–196
  - looping logic, 196–201
- logical structures, 158
- login scripts
  - modifying settings, 210–211
- looping logic, 179, 196–201
- loops
  - for, 196–198
  - C programming language, 378–379
  - testing repeatedly for condition, 199–201
  - while, 199–201
  - wildcards, 198
- ls command
  - alias, 211
  - forcing to display error message, 105
  - redirecting output, 104
  - redirecting output to more command, 142

## M

- machine language, 181
- make utility, 387–393
- make\_abs file, 389
- Makefile, 389
- make\_square file, 391
- man pages
  - copying into /man directory, 313–314
  - storing, 313
  - writing and formatting, 311–313

- manipulate and format commands, 148–154
- manipulation and transformation commands, 141
- Manycolors script, 201–202
- MEMO variable, 188
- memory usage display, 298
- menus
  - creation of, 208–210
- messages
  - displaying on-screen, 194
- methods, 406
- Microsoft FrontPage, 349
- MIME (Multipurpose Internet Mail Extensions), 280
- miscellaneous utilities, 281
- mkfs command, 284, 286
- mkfs utility, 285
- more command, 164
  - printenv command and, 184
  - redirecting ls command output to, 142
- More\_stuff script file, 150–151
- Move applet, 466–467
- moving
  - files, 111
  - windows, 426
- mv command, 111
  - alias, 211
  - I option, 111
- .myfuncs file
  - loading, 268
- .myfuncts script, 267
- MY\_VAR variable, 189–190

## N

- NAME variable
  - setting value, 188
- \$name variable, 320
- Nameaddr script, 200–201
- nested control structures, 195–196
- Netscape Composer, 349
- network utilities, 279
- Newfile1 file, 106
- Newfile3 file, 107
- Normal\_file file, 400–401
- not equal (!=) operator, 200
- nroff command, 309
- Numberlist file, 341
- numbers, 326
- numeric fields, 159
- numeric literals, 326

## O

- object code, 365
- objects, 406
- opening files, 385–386
- operands, 188
- operators, 179, 187–192
  - arithmetic, 191–192
  - defining and evaluating, 188–190
  - operands, 188
  - Perl, 323
- ordinary files, 102
- Os\_choice file, 195
  - if statement test command usage by, 241
- Os\_choice program
  - pseudocode, 227
- OS\_NAME variable, 194–195

- Our\_users file, 196
- output
  - formatting, 163–165
  - redirecting, 104
- output redirection operator (>), 106
- overwriting files
  - preventing redirection from, 192

## P

- Pact script, 168
  - adding comments, 168
  - executing, 168
- parameters
  - positional, 209
- paste command, 114–115, 164
  - d option, 115
- PATH variable, 182, 184
  - adding current working directory to, 190
  - viewing contents, 190
- Perl
  - accessing disk files, 333–336
  - accessing file by passing filename on command line, 338–340
  - alphanumeric field sort, 337–338
  - arithmetic operators, 324
  - versus* Awk programs, 330–333
  - basics, 318
  - CGI protocol, 351
  - CGI script sources, 352
  - comma (,) as list separator, 326
  - comments (#), 318, 332
  - complete statements, 319
  - decision statement, 321
  - diamond operator (<>), 332, 336

- displaying file contents, 333–335
- EOF (end-of-file) signal, 336
- eq operator, 323
- equality (==) operator, 321
- filehandles, 333
- if-else statement, 321
- information and FAQs (frequently asked questions), 352
- inline sort block, 341
- interpreting file with, 318
- logical OR (||) operator, 336
- \n (newline character), 319
- \$name variable, 320
- numeric field sort, 340–344
- opening file within program, 335–336
- operators, 323
- print statements, 319
- reading input from keyboard, 320
- relational tests on strings, 323
- sorting, 337–344
- spaceship operator, 343–344
- <STD>, 320
- STDERR file handle, 333
- STDIN file handle, 333
- STDOUT file handle, 333
- string relational operators, 324
- while loop, 330, 331
- Perl.com Web site, 352
- Perlcom.p program, 331
- Perlread.p1 program, 334–335
- Perlread.p2 program, 335–336
- Perlsort.p2 program, 338–340
- Perlsort.p3 program, 341–342
- Perlsort.p4 program, 343–344
- Perlsort.pl1 program, 337–338
- personal environment
  - customizing, 210–211
- Phlist1 program, 215–216
- Phlist1 script
  - sorting phone numbers, 269–270
- Phmenu file
  - completing, 213–214
  - determining existence of, 240
  - viewing unformatted records, 214–215
- Phmenu program, 210
  - Delete Phone Record screen, 249
  - deleting phone records, 246–250
  - record-searching capability, 245–246
- Phmenu.1 file
  - writing and formatting man page, 311–313
- phone records
  - deleting, 246–250
- Phoneadd file
  - determining existence of, 240
- Phoneadd program
  - allowing re-entry of data, 257–261
- Phoneadd script, 217–218
  - GNOME panel, 468–469
  - preventing phone number duplication, 263–265
- Phonefind program, 245–246
- Phones2 file
  - sorting, 119–120
- pipe operator (|), 142
- pipes, 142
- Pnumname script, 169
- positional parameters, 209
- POST method, 352
- pr command, 152–153
- Practivity script file, 171–172

- <PRE> tag, 350
- </PRE> tag, 350
- preprocessor, 365
- preprocessor directive, 368–369
- printenv command
  - more command and, 184
- printf function, 374–377
- printf() function, 163–165
- print() function, 132
- printing
  - three fields, 163
- processes
  - number running, 300–301
  - terminating, 301–302
- processing files, 102–106
- program development cycle, 179–182
  - high-level languages, 180–181
- programmer file, 162
  - extracting programmer name and numbers from, 169
- programmer project assignments, 160–161
- programming utilities, 280
- programs
  - redirecting input, 104
- project file, 162–163
- Project.cgi file, 352
- project.cgi file
  - source code, 357
- Project.html file, 352
- Project.html Web page
  - source code, 354–
- prototypes, 182
- prototyping applications, 182
- ps command, 300–301
- pseudocode, 227–228
  - positioning cursor, 256–257

## Q

- question mark (?) wildcard, 192

## R

- Radius.c file, 378
- rain.c file, 379–380
- rdev command, 283–284, 286
- read command, 194
- records, 103
  - allocating space in, 159
  - defining fields, 158
  - designing, 158–159
  - formatting output, 243–247
  - identifying, 159
  - layout, 158
  - numeric fields, 159
  - viewing unformatted, 214–215
- record-searching capability, 245–246
- recursively removing directories, 110–111
- redirecting input, 104
- redirecting output, 104
- redirection operators
  - retrieving input and output, 104
- regular files, 102
  - records, 103
  - trees, 103
  - unstructured ASCII characters, 103
- relational databases, 128–131
- relational integer tests, 229–231
- removing
  - duplicate lines from files, 144–145
- removing directories, 108
- removing garbage files, 293–294
- renaming files, 111

- repeating control structures, 196–201
- repositioning cursor, 256
- reusing code, 267–269
- Ritchie, Dennis, 364
- rm command, 107–108
  - alias, 210–211
  - i option, 107
  - r option, 110–111
- rmdir command, 108
- root device
  - setting, 283–284
- running applications in background, 299–300

## S

- scalars, 326
- scanf() function, 393–395
  - format specifiers, 393–394
- screen
  - clearing, 255–256
  - displaying messages on, 194
  - placing cursor at data-entry point, 208–209
- screensaver
  - selecting and configuring, 465–466
- script files, 127–128, 167–170
  - adding comments, 168
  - automatically removing temporary files, 212
  - combining, 171–172
  - comments (#), 167–168
  - creation, 127–128
  - ensuring correct shell runs, 228–229
  - executable, 128
  - executing, 168
  - joining two files, 170
- scripts
  - deleting comments, 171
- Scrmanage program, 209
- searching
  - files for pattern, 143–144
- searching Corp\_phones file, 212
- searching files
  - filenames with search string, 144
  - ignoring case, 143
  - for spaces, 144
- Secret\_file file, 400
- sed
  - appending text, 150–151
  - a\ command, 150–151
- sed command, 167, 246
- sed editor, 148–151
  - command syntax, 149
  - displaying lines and replacing text, 149–150
- select commands, 140–141, 142–148
- Seqtotal script, 193
- sequential logic, 179, 193
- servers
  - X Window System, 419
- set command
  - o noclobber option, 192
- sh command, 168
- .SH tag, 311
- shading and unshading windows, 426
- shell
  - ensuring correct runs script files, 228–229
- shell functions, 265–267
- shell operators, 187–192
- shell programs, 167–170
- shell script files, 127–128

- shell scripts, 181–182
  - executing, 181–182
  - functions, 265–267, 267, 268
  - interpreters, 181
  - making variables available to, 189–190
  - prototypes, 182
  - speed of, 182
  - test command, 229
  - user-friendly screens, 178
  - variables, 183–187
  - wildcards, 192–193
- shell variables, 179, 183
- shells, 182–183
  - logic structures, 193–203
- Simple.C file, 406
- Sim\_plus program, 406
- single quotes (‘) and spaces in variables, 188–189
- software.cgi file
  - source code, 359–360
- Software.html document, 358–359
- sort command, 118–120, 166
  - options, 119
- sort function, 338
  - as comparison function, 340
- @sortedlist variable, 338
- sorting
  - file contents, 118–120
  - Perl, 337–344
  - Phones2 file, 119–120
- sorting key, 119
- sorting phone numbers, 269–270
- sort\_name function, 269–270
- source code management utilities, 281
- Source directory, 110
- source file
  - outputting duplicate lines, 145
  - outputting unique lines, 145
- source files, 181
  - converting to machine-language, 181
  - fatal errors, 181
  - logic errors, 181
  - syntax errors, 181
- spaceship operator, 341, 343–344
- special files, 103
- spell-checking documents, 306–308
- Square\_func.c file, 391
- Square\_main.c file, 391
- .SS tag, 311
- standard error, 104
- startx command, 421
- statements
  - conditional execution, 194–196
- stderr, 104
- stdin, 104, 142
- stdio.h file, 369
- stdout, 104, 142
- strings, 194, 326–327
  - C programming language, 370–371
  - equality, 231–234
  - non-zero length, 231–234
  - testing, 231–234
  - variables as, 188–189
  - zero length, 231–234
- Stroustrup, Bjarne, 405
- subparseform.lib, 352
- syntax errors, 181
- system
  - listing most CPU-intensive tasks, 295–297
  - memory usage display, 298



- number of processes running, 300–301
- performance, 295–302
- system status utilities, 278–279, 295–302

## T

- tags, 348
- target files, 389
- temporary files
  - automatically removing, 212
- terminal emulation program
  - launching, 430
- terminating processes, 301–302
- test command, 229–243
  - a option, 236–237
  - Boolean operators, 236
  - existence of directory, 239–241
  - exit status, 230–231
  - file testing options, 234
  - integer expressions, 230–231
  - integer testing options, 229
  - o option, 237
  - relational integer tests, 229–231
  - string testing options, 232
  - string tests, 231–234
- Testfile file, 105
- Test\_file file, 234–236
  - Boolean tests, 237–239
- testing files, 234–236
- testing for end of file, 386
- testing repeatedly for condition, 199–201
- Testscript script, 190

- text
  - converting lowercase to uppercase characters, 151
  - formatting, 309–314
  - removing erroneous character from, 152
  - sed replacement of, 149–150
- text files
  - counting number of lines, bytes, or characters in, 148
- .TH tag, 310
- Thompson, Ken, 364
- TODAY variable, 189, 255
- top command, 295–297
  - forwarding display generated by, 298–299
  - options, 297
  - running in background, 299–300
  - s option, 297
- touch command, 106–107
  - a option, 107
  - c option, 107
  - m option, 107
- .TP tag, 311
- tput command, 208–209
- tput cup command
  - arguments, 209
- tr (translate utility), 243–247
  - as filter, 244
- tr command, 151–152
  - d option, 152
  - s option, 152
- trap command, 212
- trees, 103, 104
- troff command, 309

true or false values, 336  
tsch shell, 183

## U

unformatted records  
    viewing, 214–215  
uniq command, 144–145  
    -d option, 145  
    -u option, 145  
UNIX utilities, 276–281  
    classifying, 277–281  
Unix\_stuff file, 150–151  
unstructured ASCII characters, 103  
user-friendly screens, 178  
users  
    waiting for input from, 194  
USERS variable, 196  
utilities, 276–281  
    classifying, 277–281  
    communication, 280  
    file processing, 277–278, 282–294  
    miscellaneous, 281  
    network, 279  
    programming, 280  
    source code management, 281  
    system status, 278–279, 295–302

## V

values  
    true or false, 336  
variable-length records  
    cut command, 118  
    extracting fields, 117–118  
variables, 183–187, 326  
    automatic, 372  
    C programming language, 369

    clearing screen by setting,  
        255–256  
    command output as contents, 189  
    configuration, 183  
    declaring, 371–372  
    double quotes (“”) and, 194  
    double quotes (“”) and spaces  
        in, 188–189  
    environment, 183  
    global, 372  
    identifiers, 371  
    looping through range of values,  
        196–198  
    making available to shell scripts,  
        189–190  
    as operands to arithmetic  
        operators, 191  
    scalar, 326  
    scope, 372  
    setting value, 188  
    shell, 183  
    single quotes (‘’) and spaces in,  
        188–189  
    as string of characters containing  
        spaces, 188–189  
    strings, 194  
    viewing contents, 188  
variables  
    initializing, 372  
Vendors file  
    creation, 120  
    report creation from, 121–123  
Ven\_report file, 127–128  
vi editor  
    adding comments to script  
        files, 168  
    combining script files, 171–172  
    extracting programmer name and

- numbers from script, 169
- joining files, 170
- output redirection, 105
- programmer file, 162
- project file, 162–163
- reading commands from file, 105
- script file creation, 127–128
- Vmlinus file
  - finding, 285
- Vreport
  - refining, 132–133
- Vreport file, 130–131
  - generating with Awk, 131–133

## W

- Wall, Larry, 341
- wallpaper, 461–465
- wc command, 148
- Web pages
  - background and foreground colors, 350
  - body, 349
  - centering text, 350
  - CGI (Common Gateway Interface), 351–357
  - creation of, 349–350
  - head, 349
  - headings, 350
  - hyperlinks, 348
  - indenting text, 350
  - preformatted text, 350
  - setting up, 348
  - testing, 360
  - UNIX sample, 352–357
- Web servers, 348
- while loop

- C programming language,
  - 378–379
  - as data entry form, 200–201
  - test command usage, 242–243
- while loops, 199–201
- wildcards
  - cp command and, 109
  - deleting files, 108
  - grep command, 143–144
  - loops, 198
  - shell scripts, 192–193
- Window Managers, 420
- windows, 423
  - borders, 424
  - Close button, 424, 427
  - Gnome Help Browser button, 427
  - help pop-up boxes, 427
  - Iconify button, 424, 427
  - interacting with, 424–428
  - Minimize/Maximize button, 424, 427
  - moving, 426
  - resizing, 425–426
  - scrollbar, 424
  - shading and unshading, 426
  - title bar, 424
  - Window Options button, 424
  - Windows Options button, 428
- worldwidemart.com Web site, 352

## X

- X clients, 419
- X servers, 419
- X Window System
  - adding applications to main menu, 470–471

- automatically starting, 422–423
- Calendar application, 446–453
- clients, 419
- components, 423
- configuring desktop, 461–471
- configuring Linux to start, 421–422
- defining, 418
- Desktop area, 423
- GNOME (GNU Network Object Model Environment), 420
- GNOME panel, 423, 428–432
- icons, 423
- interacting with, 423–433
- interacting with windows, 424–428
- multiple terminal windows in production environment, 458–461

- running File Manager, 437–446
- servers, 419
- Spreadsheet application, 453–458
- starting, 421–423
- Window Managers, 420
- windows, 423
- X11, 418
- XFree86, 418
- X-Window System
  - executing Netscape from, 353–354
  - opening Terminal window, 353
  - starting, 352

## Z

- zch shell, 183